FINITE ELEMENT SIMULATIONS OF COMPREHENSIVE MITRAL VALVE MODEL

A Thesis
Presented to
The Academic Faculty

by

Marcel Pena

In Partial Fulfillment
of the Requirements for the Degree
B.S. in Biomedical Engineering with Research Option in the
School of Biomedical Engineering

Georgia Institute of Technology
December 2014

FINITE ELEMENT SIMULATIONS OF COMPREHENSIVE MITRAL VALVE MODEL

Approved by:

Dr. Milan Toma, Advisor
School of Biomedical Engineering
Georgia Institute of Technology

Dr. Ajit Yoganathan
School of Biomedical Engineering
Georgia Institute of Technology

Dr. Essy Behravesh
School of Biomedical Engineering
Georgia Institute of Technology

Date Approved: November 20, 2014

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS AND ABBREVIATIONS

MR	Mitral Regurgitation
MV	Mitral Valve
FSI	Fluid-structure interaction
CFD	Computational fluid dynamics
SPH	Smoothed Particle Hydrodynamics
CPU	Computer Processing Unit
GPU	Graphics Processing Unit
SPH	Smoothed Particle Hydrodynamics
FEA	Finite Element Analysis
FE	Finite Element
LC	Characteristic Length
IMPETUS	IMPETUS Afea Solver

SUMMARY

Mitral regurgitation (MR) is an increasingly prevalent disorder of the heart in which the mitral valve (MV) does not close properly causing blood to flow backward (leak) into the upper heart chamber when the left lower heart chamber contracts. MR is the most common type of heart valve insufficiency, often requiring open-heart surgical repair. After age 55, some degree of MR is found in almost 20% of men and women. MV repair is considered superior to mitral valve replacement, and there are many surgical techniques utilized to address differing pathologies. One approach to assess the effects of pathology and proposed surgical repair is to utilize a computational model, in which pathologic or surgical alterations can be assessed systematically. However, current models are limited by assumptions related to geometry and material properties and importantly, none have been validated with detailed experimental data. In the present work, an advanced fluid-structure interaction (FSI) model of the MV system is utilized without the limitation of geometry and material properties. This model allows analysis of the valve in the normal, diseased, or repaired states.

CHAPTER 1

INTRODUCTION

Mitral regurgitation is the most common heart valve problem encountered in clinical practice [1]. It is a disorder of the heart in which the mitral valve does not close properly when the heart pumps out blood. Thus there is an abnormal leaking of blood from the left ventricle, through the mitral valve, and into the left atrium, when the left ventricle contracts. Mitral valve repair is considered superior to mitral valve replacement [2], and there many surgical techniques utilized to address differing pathologies. Not all damaged valves are suitable for repair; in some, the state of valve disease is too advanced and replacement is necessary. There has been a great debate about the timing of surgery in patients with asymptomatic mitral valve regurgitation. One approach to assess the effects of pathology and proposed surgical repair is to utilize a computational model, in which pathologic or surgical alterations can be assessed systematically. However, current models are limited by assumptions related to geometry and material properties [3] and importantly, none have been validated with detailed experimental data. In this study, we utilize an advanced FSI model of the mitral valve that allows us to check the convergence of the solution. The solution from the finite element program is checked with a solution of increased accuracy. In this study we use three different meshes of different accuracy. After we prove in this study that a solution has converged, this FSI model can be utilized to assess many different types of pathology and repair [4] [5] by analyzing the direction of blood flow, blood flow rate, blood flow volume and blood pressure while simultaneously simulating the influence of blood flow on the surrounding structures of the mitral valve. The ultimate long-term goal of this research is to provide an advanced

fluid-structure interaction model of the mitral valve that could ultimately be used for individualized patient planning for mitral valve repair.

Literature Review

Profound anatomical and pathophysiological knowledge of the MV and adjacent cardiac structures are essential for selecting the most appropriate surgical techniques in MV disease, especially with regards to modern minimally-invasive MV surgery [6, 7]. Imaging is the key to study the lesions and dysfunction of MV. With regards to the complexity of the MV apparatus, defined as nonlinear system of FSI, is at present not fully understood. Exact modeling of MV function has not been realized in the past due to the lack of imaging techniques that enable detection of the 3D structures and their respective function. With the development of real-time 3D echocardiography and 4D computed tomography, however, the technical capabilities have been introduced to create substantial and well-defined MV models with a high probability for implementation into clinical practice [8]. Recent studies have developed a novel evaluation protocol to create a virtual MV model using 3D TEE data in a patient with posterior partial ruptured mitral chordae tendineae (RMCT) to determine the effect of a virtual repair using expanded polytetrafluoroethylene (ePTFE) sutures [2].

These models may be utilized to create simulation techniques, which will hopefully allow for much better operative outcomes. Additionally, recent work by Rim et al. [9] shows that by using 3D-TEE to obtain volumetric morphology of the mitral valve apparatus, additional computational techniques can provide virtual mitral valve models that demonstrate deformation and stress distribution on the mitral valve structure across

the cardiac cycle. Besides the ones mentioned, many other finite elements studies have been done to asses different pathologies of diseased MV such as MV stenosis and percutaneous MV dilation [10] and regurgitation [11-14] with prospective repair options. Due to the large deformations, pulsatile loads during the cardiac cycle and anisotropic non-linear elastic behavior of the valvular tissue, biomechanics of the mitral valve is a highly complex problem. For the sake of efficiency, the current computational models in the literature often use simplified geometry, e.g. the published studies commonly take advantage of the chordal beam-type structure to justify the use of beam elements [7, 11, 12] and the thinness of the leaflets to justify the use of shell elements [2, 5, 15]. Some groups do model the leaflets using 3D elements, e.g. tetrahedral, but with insufficient refinement, such as single layer, in order to save the computational time, thus sacrificing the accuracy of the computations [14, 16]. None of the currently published studies consider chordal structure as the 3D part of the model. These authors, point out that being able to assess stress distribution on the mitral leaflets, even in patients who do no currently demonstrate severe MR, may allow identification of patients who are most likely to develop pathophysiologic alterations in the mitral valve structure. Thus, they will be able to predict in these patients who may be more likely to develop potential chordal ruptures or annular dilations. The ability to determine the exact location of stress distribution on pathologic mitral valve leaflets should provide the surgeon with a roadmap to better plan operative intervention, one that might prevent technical failures in the future.

CHAPTER 2

MATERIALS AND METHODS

Convergence

When modeling a problem using a finite element program, it is very important to check the validity of the solution. Checking on the convergence of the solution does this. Convergence is tested differently depending on which solution method is used. The two are the p-refinement and the h-refinement.

The word convergence is used because the output from the finite element program is converging on a single correct solution. To check the convergence of the solution, at least two solutions to the same problem are required. The solution from the finite element program is checked with a solution of increase accuracy. If the more accurate solution is dramatically different from the original solution, then the solution is not converged. However, if the solution does not change much (less than a few percent difference) then the solution is considered converged. In this study we use the h-refinement and the p-refinement to check on convergence.

Fluid-structure interaction

Fluid-structure models aim to study the interaction between the MV and the blood flow. In such models, a structural model of the finite element model (FEM) is combined with a computational fluid dynamic (CFD) model. This allows for analysis of flow and dynamic structures. Fluid-structure interactions can be stable or oscillatory. In oscillatory

interactions, the strain induced in the solid structure causes it to move such that the source of strain is reduced, and the structure returns to its former state only for the process to repeat.

Mesh generation

A *mesh* is a discretization of a geometric domain into small simple shapes, such triangles or quadrilaterals in two dimensions and tetrahedral or hexahedra in three. Meshes find use in many application areas. In cardiovascular medicine, one can distinguish between different types of meshes: structured, unstructured and hybrid. Structured grids are identified by regular connectivity, and unstructured by irregular connectivity. A hybrid mesh contains a mixture of structured portions and unstructured portions. Exact modeling of MV function has not been realized in the past due to the lack of imaging techniques that enable detection of the 3D structures and their respective function. With the development of real-time 3D echocardiography and 4-dimensional computed tomography, however, the technical capabilities have been introduced to create substantial and well-defined MV models with high probability for implementation into clinical practice.

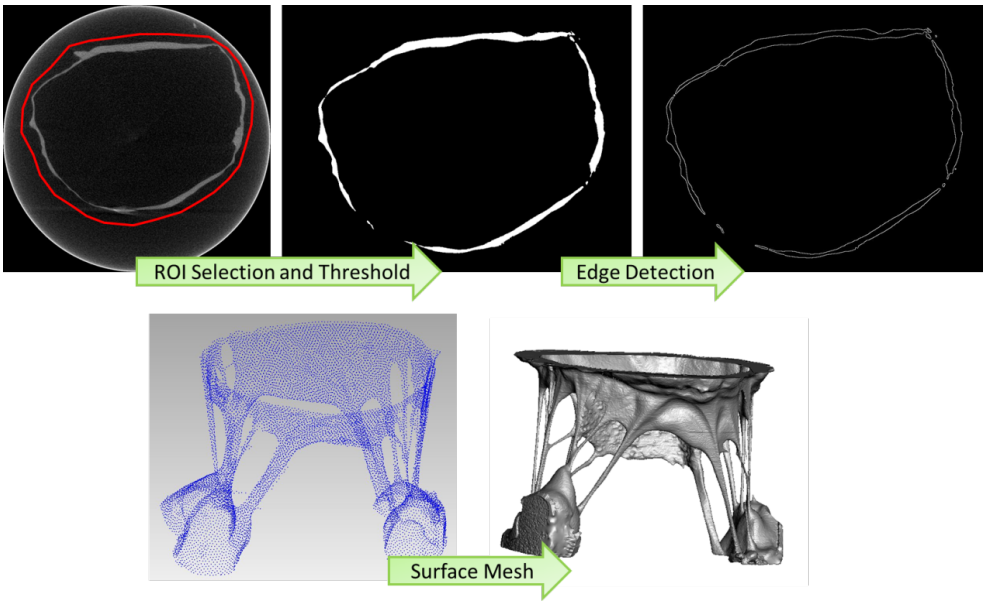

Figure 1. The Mitral Valve Model. Process of taking a patient-specific MV structure into a FEM.

Dr. Einstein from Pacific Northwest National Laboratory generated an Imaging-based model (see **Fig. 1)** of the MV by using a micro-CT scan that provided very high-resolution 3D images of the MV, and showed all the components including the chords and the insertion points into the leaflet, with up to 39 microns resolution. Additionally, he also used a 3D echo that obtained temporal measurements through the cardiac cycle. An in house segmentation code was capable of providing very accurate leaflet morphology, while accurately identifying the individual leaflets of the MV. For this study we replicated the MV model three times, and each time with a 10, 20, and 40% increase in the number of tetrahedral.

GPU Technology

The Graphics Processing Unit (GPU) is used more and more for Scientific Computing. For relatively low costs one can obtain supercomputer performance (1 Teraflop). There are several solvers that are adapted in order to make them suitable for GPU computing. We looked to improve turnaround times for the complex FSI simulations performed on each greatly refined MV meshes. Only recently have GPUs been specifically developed with the computational precision required for finite element analysis (FEA) solutions as well as the computational power to effectively complement the performance of the latest CPUs. With hundreds of low-power cores on a single socket, GPUs had the potential to dramatically increase computing capacity, provided that the computed workload will fit in the available memory of the GPU. The accelerator works by offloading some of the most numeric-intensive algorithms from the CPU onto the GPU. These algorithms are part of the equation solutions occurring during a simulation.

IMPETUS Afea Solver®

IMPETUS Afea Solver® (IMPETUS) is a software package for non-linear computational mechanics. The package comprises four different solver modules and Post Processor.

Finite Element (FE) module: The FE module is a non-linear explicit finite element code, primarily developed to predict large deformations of structures and components exposed to extreme loading conditions. The module has a material library that covers most solid material types, such as metals (isotropic and anisotropic), rubber, ceramics, fiber composites, foams and soil. The module is based on a unique volume element technology with focus on accuracy and robustness. The FE module comes together with an advanced Post Processor for visualization of the simulation results.

Discrete Particle module: This module is used to describe the physical behavior of sand, air, and high-explosives by discrete particles. The module adds to and communicates with the FE module in order to fully couple the structural response with the behavior of the material represented by particles. Several explosive materials as well as wet and dry sand conditions have been predefined.

WELDSIM module: WELDSIM is a finite element code specialized in simulations of welding processes. It consists of a thermal, a micro-structure and a mechanical sub-model, that are coupled sequentially. By coupling the thermal solver to a micromechanical model several parameters such as yield stress distribution in the HAZ can be determined. The FE module can directly utilize the results from WELDSIM, allowing large deformation analysis to be run with correct material properties in the HAZ.

Smoothed Particle Hydrodynamics (SPH) module: This module is used to describe fluids, such as water. The module adds to and communicates with the FE module in order to obtain fully coupled simulations through a fluid-structure interaction interface. The implementation enables a high scalability on GPU machines and allows 20,000,000 SPH elements models on a standard desktop computer.

For the purposes of this research project, namely the FSI analysis, two modules are used to perform the simulations, i.e. the FE module and SPH module.

H-refinement and P-refinement

In order to provide a comprehensive convergence analysis, the h-refinement and the p-refinement are utilized. In order to increase the accuracy of the solution, the complexity of the shape function must be increased (p-refinement) or more elements must be added (h-refinement).

H-refinement

The simplest type of element has a linear shape function. This means that the function for displacement across the element is linear. With the h-refinement, the shape function of the element will usually be linear. In an actual part, it is quite uncommon for the displacement to vary linearly. The h-refinement accounts for this by increasing the number of elements. More accurate information is obtained by increasing the number of elements.

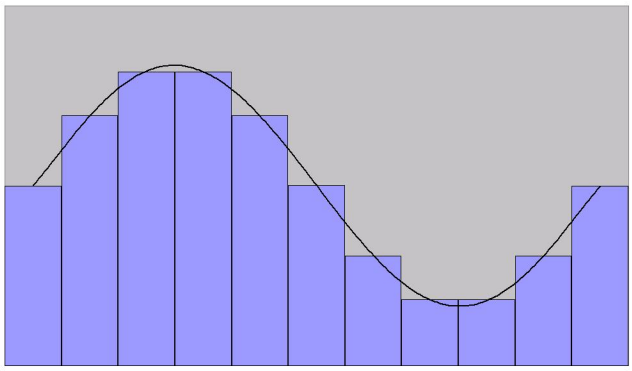

Figure 2. H-refinement with a coarse mesh.

It is supposed that the actual stress across a part varied by the function represented by the curve in **Fig. 2**. If the problem were analyzed using linear shape functions, then the bars in **Fig. 2** would represent the results for a coarse mesh.

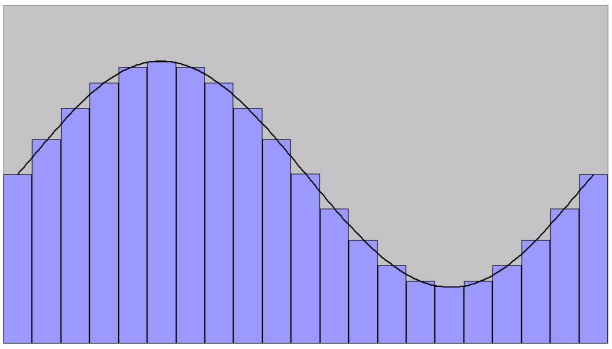

Figure 3. H-refinement with a fine mesh.

If a part is modeled with a very coarse mesh, then the stress distribution across the part will be very inaccurate. In order to more accurately find the stress distribution across the part, it is needed to increase the number of elements. If the number of elements are doubled, then the stress distribution would be represented by the bars in **Fig. 3.**

P-refinement

With p-refinement, once a mesh is created, it does not need to be changed. Rather than changing the number of elements, the shape function of the element will be changed to handle non-linear displacement functions. In areas where the stress is changing quickly, the complexity of the shape function is changed rather than changing the size of

the elements. More accurate information is obtained by increasing the complexity of the shape function.

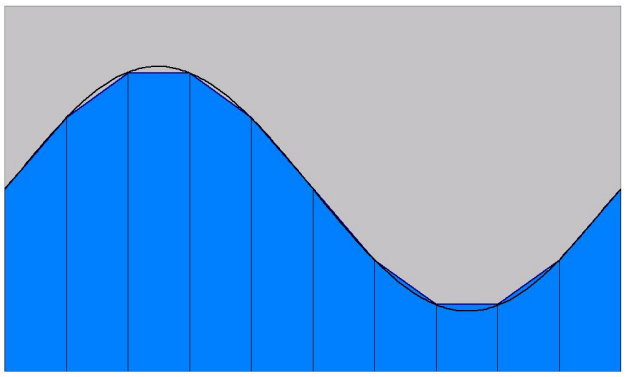

Figure 4. P-refinement with second order polynomial.

The p in p-refinement stands for polynomial. Increasing the polynomial order of the shape function changes the accuracy of the p-refinement. This allows a very complex displacement function to be approximated across a large element.

If the shape function is second order, then the strain across the element will be linear. Using the same example stress distribution, as before, the bars in **Fig. 4** would represent the results.

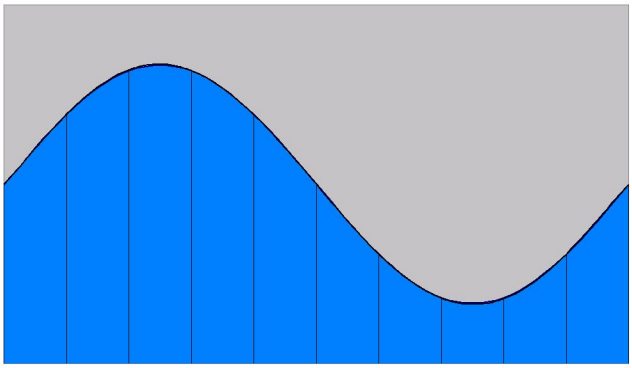

Figure 5. P-refinement with a 3rd order polynomial.

If this does not accurately reflect the strain in the element, then the order of the shape function can be increased to third order. This will allow strain over the element to be a second order function. The bars in **Fig. 5** would represent the results.

GMSH

In order to implement the h-refinement, the GMSH 3D finite element grid generator is used.

GMSH is a 3D finite element grid generator with a built-in CAD engine and post-processor. This mesh generator is chosen because its design goal is to provide fast, light and user-friendly meshing tool with parametric input and advanced visualization capabilities. GMSH is built around four modules: geometry, mesh, solver and post-processing.

The geometry module is used to create the geometrical entities, which are called 'elementary' in the GMSH's jargon and are assigned identification numbers (strictly positive) when they are created:

1. Each elementary point must possess a unique identification number;
2. Each elementary line must possess a unique identification number;
3. Each elementary surface must possess a unique identification number;
4. Each elementary volume must possess a unique identification number;

Geometry Commands

`Point (`*`expression`*`) = {`*`expression`*`, `*`expression`*`, `*`expression`*` <, `*`expression`*` >};` **(1)**

Command **(1)** creates an elementary point. The expression inside the parentheses is the point's identification number; the three first expressions inside the braces on the right hand side give the three X, Y and Z coordinates of the point in the three-dimensional Euclidean space; the optional last expression sets the prescribed mesh element size at that point. See *Specifying mesh element sizes*, for more information about how this value is used in the meshing process.

```
Line (expression) = {expression, expression};                    (2)
```

Command **(2)** creates a straight line segment. The expression inside the parentheses is the line segment's identification number; the two expressions inside the braces on the right hand side give identification numbers of the start and end points of the segment.

```
Line Loop (expression) = {expression-list};                      (3)
```

Command **(3)** creates an oriented line loop. The expression inside the parentheses is the line loop's identification number; the expression-list on the right hand side should contain the identification numbers of all the elementary lines that constitute the line loop. A line loop must be a closed loop, and the elementary lines should be ordered and oriented (using negative identification numbers to specify reverse orientation). If the orientation is correct, but the ordering is wrong, GSMH will actually reorder the list internally to create a consistent loop.

```
Plane Surface (expression) = {expression-list};                  (4)
```

Command **(4)** creates a plane surface. The expression inside the parentheses is the plane surface's identification number; the expression-list on the right hand side should contain the identification numbers of all the line loops defining the surface.

```
Surface Loop (expression) = {expression-list};
```
(5)

Command **(5)** creates a surface loop (a shell). The expression inside the parentheses is the surface loop's identification number; the expression-list on the right hand side should contain the identification numbers of all the elementary surfaces that constitute the surface loop. A surface loop must always represent a closed shell, and the elementary surfaces should be oriented consistently (using negative identification numbers to specify reverse orientation).

```
Volume (expression) = {expression-list};
```
(6)

Command **(6)** creates a volume. The expression inside the parentheses is the volume's identification number; the expression-list on the right hand side should contain the identification numbers of all the surface loops defining the volume.

MV: Tissue material properties

The tissue material properties of the MV are defined in *Tissue_materials_model.k.* In this file, a collection of parameters listed in **Table 2** named *mat_tissue_dispersed* has been specifically generated for the purpose of this research project and added to the IMPETUS library. The rest of this section is an attempt to explain the model used to assign an orientation to each MV fiber as the MV closure occurs.

Conceptually, the idea is that collagen is statistically oriented in tissues, having an overall mean orientation with dispersion or splay. It is common to think of this dispersion in terms of a probability density function, having both mean and variance. This dispersion may be only in the plane or it may be three-dimensional. Strictly speaking, the base model (TYPE=1) is also orthotropic, if the dispersion has a non-zero variance. The type of orthotropic depends on whether or not the user selects the option to disperse fibers in the 3rd dimension. Number_of_fibers=2 considers two families of fibers, symmetrically distributed about a mean angle. The fiber coordinate system for each integration point is specified in the *INTIAL_TEXTURE keyword, and is given in global coordinates, such that the first vector represents the mean fiber orientation and the second vector is in the tissue plane. Thus, the cross-product of the two vectors represents the tissue-plane normal, typically the element face normal. If TEXTURE=1, three vectors and magnitudes are given in the *INTIAL_TEXTURE keyword, such that dispersion can be assigned directly from medical imaging data[15].

The following section describing the materials properties of the MV is based on [17, 18]:

Variable	Description
mid	Unique material identification number
$\boldsymbol{\rho}$	Density
$\boldsymbol{\mu}$	μ is the isotropic shear modulus that models elastin. μ may be zero, if and only if $\zeta \geq 0$ and $f \neq 0$
$\boldsymbol{\zeta}$	Governs the extent of dispersion. As ζ goes to zero, the material symmetry reduces to pure transverse isotropy. Conversely, as ζ becomes large, the material symmetry becomes isotropic in the plane.
$\boldsymbol{f}$	Parameter f $(0 \leq f \leq 1)$ governs the extent to which fiber dispersion extends to the third dimension. $f = 0$ and $f = 1$ apply to 2D splay with the normal to the membrane being in the β and the γ-directions, respectively. $f = 0.5$ applies for 3D splay with transverse isotropy. Splay will be orthotropic whenever $f \neq 0.5$.
$\boldsymbol{\kappa}$	Bulk modulus
TYPE	1D collagen type: exponential (TYPE=1) or coiled (TYPE =2)
A	First parameter in exponential fiber model
B	Second parameter in exponential fiber model
$\boldsymbol{H_0/r_0}$	Coiled fiber model: initial wavelength of crimp (H_0) divided by the initial fiber radius (r_0)
Number_of_fibers	1 for single family of fibers (e.g. skin, tendon or valve tissue), 2 for two family of fibers (e.g. blood vessels)
$\boldsymbol{\theta}$	Angular separation between fiber families, if number_of_fibers = 2
TEXTURE	Indicates whether initial texture vectors are unit vectors (default) or if the magnitudes are specified by the *INITIAL_TEXTURE keyword

Table 2. MAT_TISSUE_DISPERSED parameter definition. Impetus uses these parameters to simulate the tissue materials properties of the MV. in the file *Tissue_material_models.k.*

The stress in the reference configuration (2nd Piola-Kirchhoff) consists of one matrix term, a hydrostatic pressure term and either one (default) or two (number_of_fibers=2) fiber terms:

$$\boldsymbol{S} = \kappa J(J-1)C^{-1} + \mu J^{-2/3}\boldsymbol{DEV}\left[\frac{1}{4}(I - \bar{C}^{-2})\right] + \sum_{i=1}^{n} \sigma_i(\lambda_i)J^{-2/3}\,\boldsymbol{DEV}[\boldsymbol{K}_i] \quad \textbf{(1)}$$

Where S is the second Piola–Kirchhoff stress, J is the Jacobian and κ is the bulk modulus; σ_i is the passive fiber stress–strain rule for the ith (1 or 2) fiber population, and DEV is the deviatoric projection operator:

$$DEV[\bullet] = (\bullet) - \frac{1}{3}tr\big((\bullet)\mathrm{C}\big)C^{-1} \quad \textbf{(2)}$$

And C is the right-Cauchy deformation. The dispersed fourth invariant is analogous to the fourth invariant from the classic theory by Spencer [1973]:

$$\lambda_i{}^2 = tr\left[K\overline{C}\right] \quad \textbf{(3)}$$

Where $\overline{C}$ is the isochoric part of the right Cauchy deformation. Note that, λ is not a stretch in a classical sense, as **K** embodies the concept of dispersion. **K** is called the dispersion tensor or anisotropy tensor and is given in global coordinates. The fiber models are defined in the fiber coordinate system. In effect, **K** rotates and weights these 1D models, such that they are both three-dimensional and in the Cartesian frame.

K is given by:

$$K_i = Q_i k_i Q_i^T \quad \textbf{(4)}$$

$$K_i = Q_i \begin{bmatrix} \frac{1}{2}(1 + e^{-2\zeta_i{}^2}) & 0 & 0 \\ 0 & \frac{f_i}{2}(1 - e^{-2\zeta_i{}^2}) & 0 \\ 0 & 0 & \frac{1-f_i}{2}(1 - e^{-2\zeta_i{}^2}) \end{bmatrix} Q_i^T \quad \textbf{(5)}$$

Where Q_i is the transformation tensor that rotates k_i from local coordinates to global Cartesian coordinates.

In the case where TEXTURE = 1, K is given by:

$$K_i = Q_i^T k_i Q_i \quad \textbf{(6)}$$

$$K_i = Q_i^T \begin{bmatrix} X_1 & 0 & 0 \\ 0 & X_2 & 0 \\ 0 & 0 & X_3 \end{bmatrix} Q_i \quad \textbf{(7)}$$

In this case, the transformation goes the other way since the X values are given in a global not local coordinate system.

Currently, there are two choices of fiber models, an exponential model (TYPE= 1) and a coiled collagen model (TYPE = 2). The exponential model is given by:

$$\sigma = C_1 \left(e^{\frac{C_2}{2}(\lambda^2 - 1)} - 1 \right) \quad \textbf{(8)}$$

CHAPTER 3

RESULTS

GMSH

The original MV is comprised by 65,080 vertices, defined as points in 3D space with (x,y,z) coordinates. A MATLAB algorithm iterates through the coordinates file, copies each point and write the point in a new file using the GMSH format, resulting in **Fig. 6**.

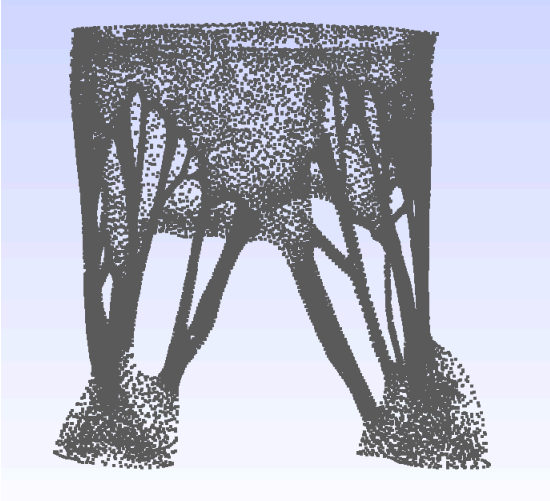

Figure 6. Mitral Valve vertices. Picture obtained from the GMSH platform after writing the MV coordinates in GMSH format.

Once the points are written in the GMSH format, the next step is to connect them with straight lines. The algorithm *connectLines.m* is accomplished by iterating through each tetrahedral element defined in the elements file. Four nodes, and six straight lines comprise each tetrahedral element, as shown in **Fig. 7**. The task of the algorithm is to define a straight line between each node, connecting the four nodes in each of the 293,000 tetrahedral elements. **Fig. 8** shows all the points comprising the anterior leaflet fully connected by straight lines.

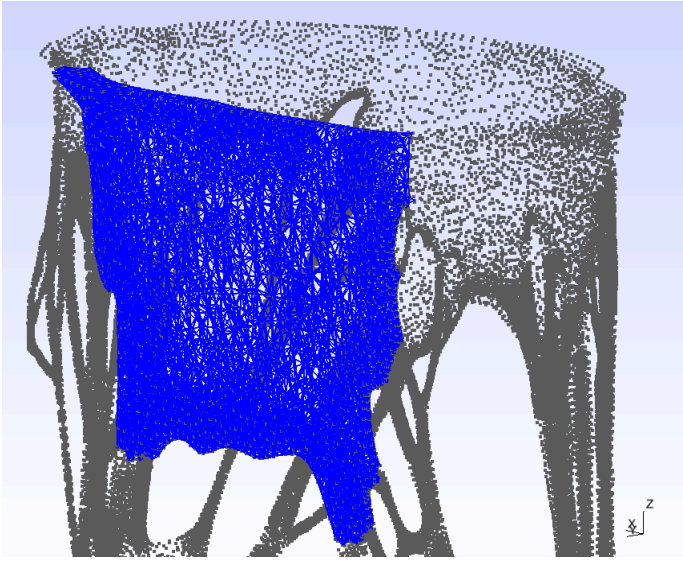

Figure 8. Anterior leaflet nodes connected with straight lines. Picture obtained from the GMSH platform after a MATLAB algorithm defined straight lines between the nodes comprising one of the leaflets.

The final step is to define a closed loop of straight lines, which allows defining the triangular surface of the tetrahedral. Finally, a loop of four surfaces allows defining a tetrahedral element as a volume. Once the 293,000 tetrahedral elements of the FEM are fully defined as volumes, the Mesh module is used.

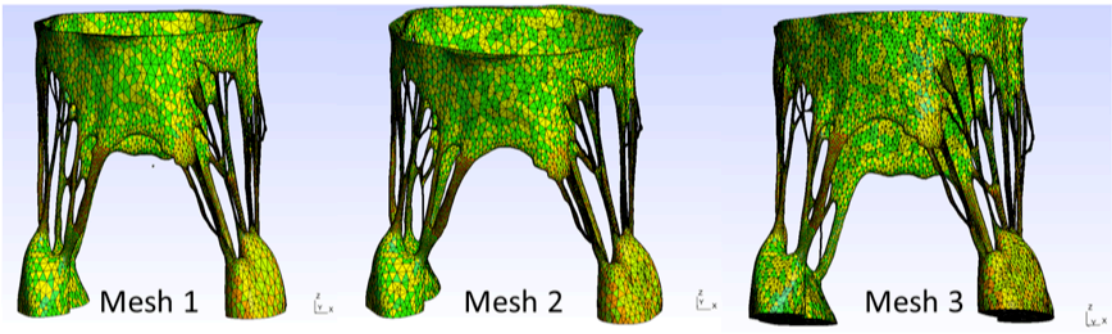

Figure 9. H-refinement results in GMSH. Mesh 1 has the original number of elements. Mesh 2 and 3 are refined with an increase in 20 and 40% from Mesh 1. It can be noticed how the tetrahedral elements become smaller in Mesh 2 and Mesh 3 compared to Mesh 1.

Once a fully defined FEM is obtained from GMSH, the *Mesh* module is used to produce two more refined meshes. The characteristic length (lc), which sets the target element size at the point, is modified for this purpose. By trial and error, two meshes with the target number of elements are obtained, with an increase in 20% and 40 % for mesh 2 and mesh 3 respectively, as shown in **Fig. 9. Table 1** summarizes the final number of elements for each mesh. The characteristic length for Mesh 1 represents the original MV mesh without extra elements.

Name	Number of elements	Characteristic Length
Mesh 1	293,000	1
Mesh 2	351,600	0.040
Mesh 3	410,200	0.043

Table 1. Mesh Element distribution. The characteristic length (lc) sets the target element size at the point. The target number of elements defined for Mesh 2 and Mesh 3 were calculated as a baseline number of elements plus an increase in 20% and 40% of the baseline, for mesh 2 and mesh 3 respectively.

Impetus: Initial and Boundary Conditions

The MV mesh is comprised by a total of 72 parts. Each part represents an anatomical section of the MV i.e. Chords, papillary muscles, leaflets and annulus. To simulate the closure of the MV under fluid interaction, boundary and initial conditions had to be defined.

The MV is submerged in a pipe filled with a discrete number (N) of fluid particles with a density ρ. To generate a pressure gradient on the outer walls of the leaflets, two pistons are located at both ends of the pipe, which can be identified by the gray tap (piston 1) and green tap (piston 2). As the bottom piston moves up at a velocity described in (Fig. 8) the upper piston also follows at a velocity described in **Fig. 11**, the difference in velocity between the bottom piston and the top one, produces an increase in pressure on the outer walls of the leaflets, hence causing the leaflets to come together and close the MV.

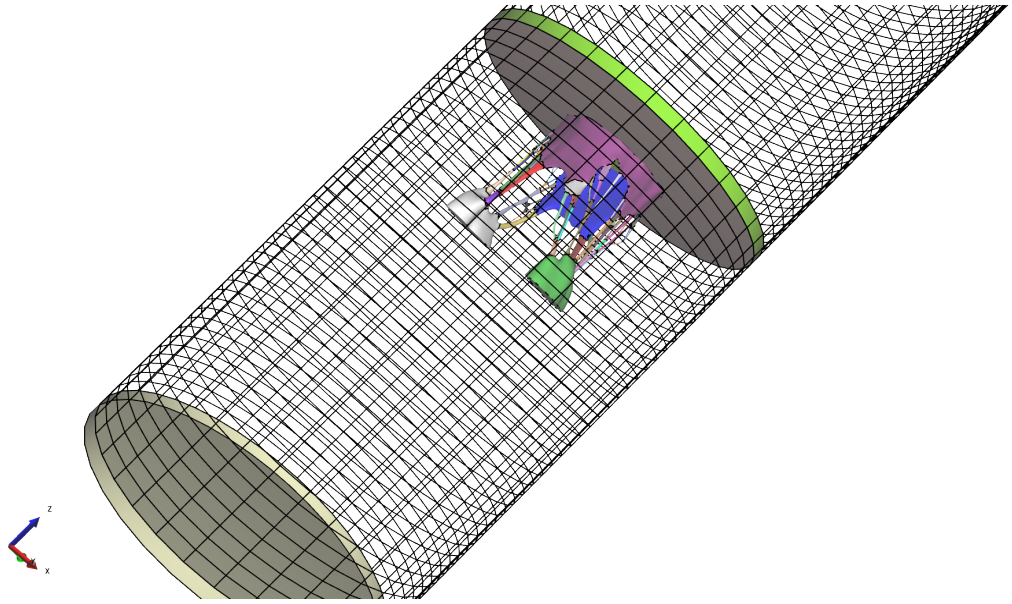

Figure 10. MV simulation assembly. Piston one is located at the ventricular site, and piston two at the atrial site. A plate is merged to the annulus of the MV shown as the bottom of piston 2 in gray color, so fluid is driven through the valve. Both pistons move at a different velocity, causing a change in pressure to occur.

Pupillary muscles, the pipe, both pistons and the plate are defined to behave as rigid and fixed, or not deformable under pressure. IMPETUS takes this input in the parameter *mat_rigid*, which takes the ID of the part as well as a density $\rho = 1000\,\frac{kg}{m^3}$.

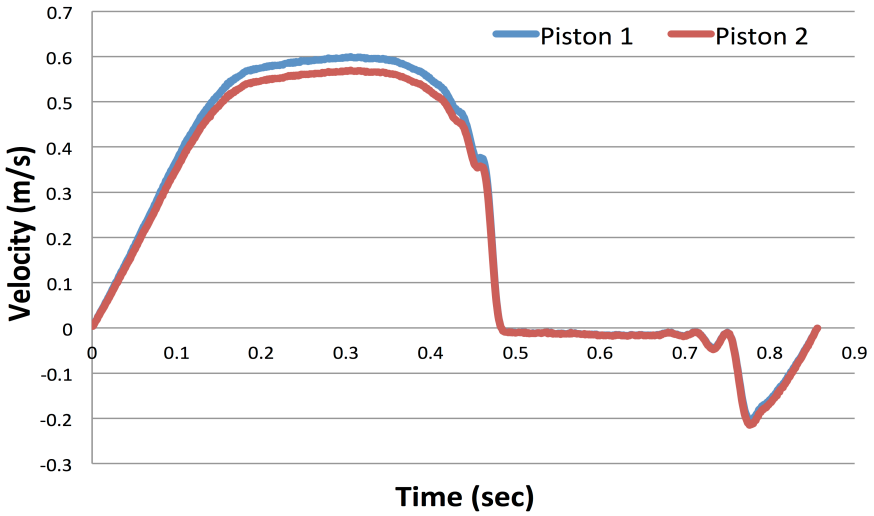

Figure 11. Piston 1 and 2 velocity profiles. Piston 1 overshoots higher than piston 2 to produce a pressure gradient which pushes the outer walls of both

To keep the MV in place during the simulation, the plate is attached to the pipe as well as to the annulus of the MV; by assigning to the merging surfaces with the same ID number. Moreover the task of the plate is to direct the fluid through the valve rather than around it, representing the function of the MV in the heart.

The prescribed motion of both pistons is defined in IMPETUS through the parameter *BC_motion* (**Table 2**). In this parameter the kinematic boundary conditions are defined. When referring to a local cylindrical coordinate system, X corresponds to the radial direction, Y to the tangential direction, and Z to the axial direction.

MV: Tissue material properties

Equations from 1-8 provides to each fiber an orientation as the MV closes. The orientation is based on the material properties of the fiber.

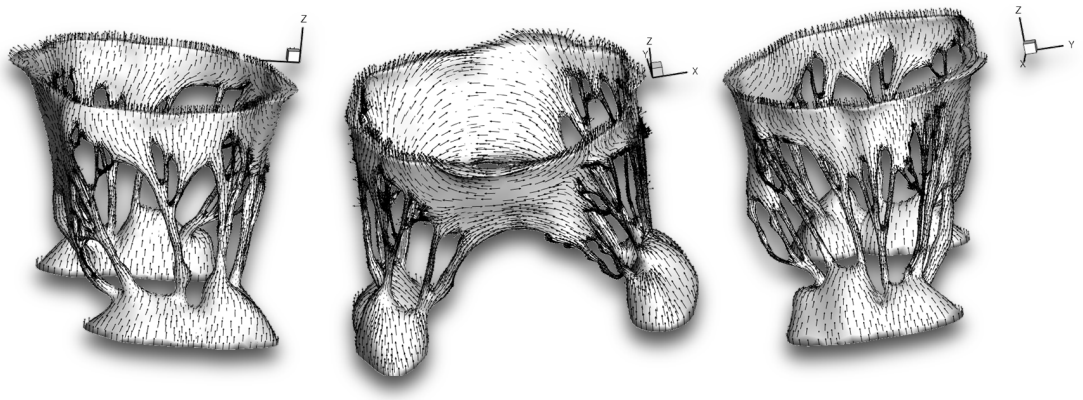

Figure 12. Fiber Orientation Model. Equations from 1-8 describe the orientation of the fiber. The orientation of the fibers in the leaflets show the direction in which they close. In the chords, the fibers have an upward orientation, as they stretch.

Convergence Criteria

To measure the accuracy of each simulation a reference value, $\boldsymbol{D_{ref}}$, is taken and considered to be the most accurate solution that can be obtained. $\boldsymbol{D_{ref}}$ is computed using the highest p-order i.e. cubic polynomial order, and the mesh with the highest number elements i.e. Mesh 3. It is defined that a solution of a simulation has converged, when it is within a 2% deviation from $\boldsymbol{D_{ref}}$.

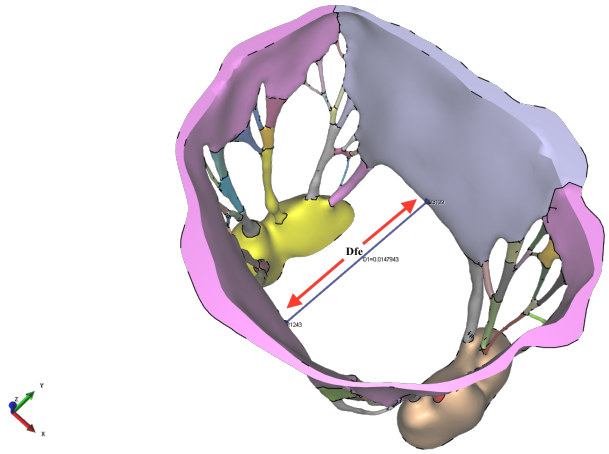

Figure 13. Measured D_{fe} in the Mitral Valve. The displacement D_{fe} can be seen as a line connecting two points located on opposite leaflets.

D_{fe} (See **Fig. 13**) is a dependent variable, depends on the refinement of the mesh as well as the p-order selected. D_{fe} is the displacement between two points randomly picked on each leaflet. As the MV closes, the two leaflets come closer together resulting in a displacement D_{fe}. At the same time point, a D_{fe} is recorded for each of the nine simulations.

P and H-refinement

In **Fig. 14** as the logarithm function is evaluated at points that are closer and closer to zero, the value of the function at those points becomes more and more negative. In **Fig. 12**, the y-axis depicts a logarithm function that takes a ratio $\frac{D_{fe}}{D_{ref}}$ and subtracts it from 1. Each point in **Fig. 14** represents a D_{fe} value recorded. The dashed line in **Fig. 15**, drawn at -4 represents a value of D_{fe} which is 98% of the reference value D_{ref}, or $ln\left(1-\frac{D_{fe}}{D_{ref}}\right)=-4$.

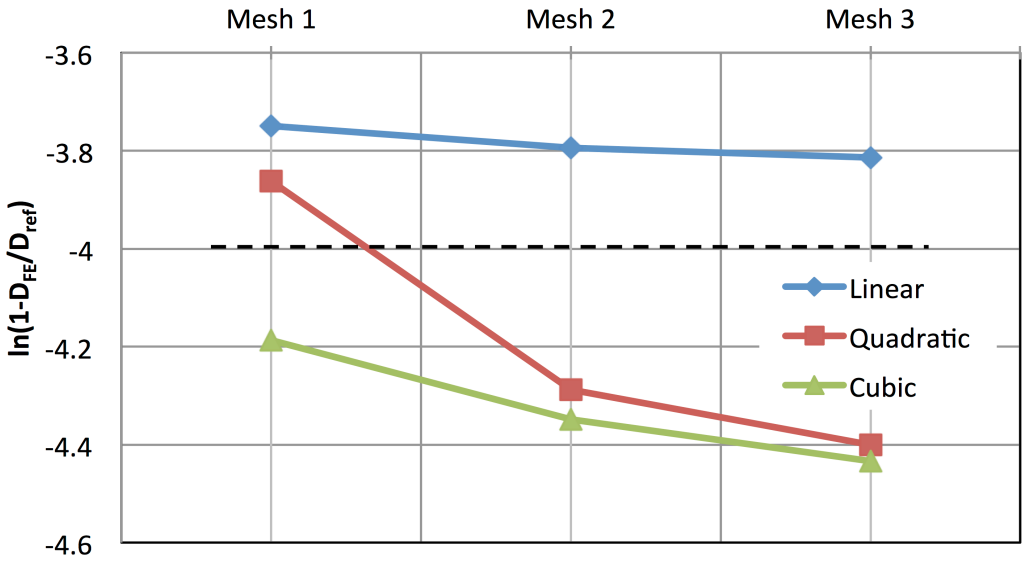

Figure 14. P/H-refinement solution convergence analysis. P and H-refinement were used to assess the solution convergence of a human mitral valve finite element model. The x-axis represents Mesh 1, 2 and 3. The y-axis represents the percentage change from the reference value. For each mesh, a linear, quadratic and cubic polynomial solution are computed. The dashed line represents where $\boldsymbol{D_{fe}}$ is 98% of $\boldsymbol{D_{Ref}}$.

In **Fig. 14** as $\boldsymbol{D_{fe}}$ approaches to $\boldsymbol{D_{ref}}$, a more accurate solution is obtained, resulting in the logarithm function to become more negative. Each mesh represents an increase in the number of elements, which are refined using the GMSH platform. Increase in the p- and h-refinement yielded a more accurate solution.

CHAPTER 4

DISCUSSION

It is shown in **Fig. 14** that for the linear polynomial order, none of the simulations yielded a solution below the dash line. On the other hand, for the quadratic order polynomial solutions, only Mesh 2 and 3 converged. Finally, all the solutions using a cubic order polynomial converged.

It is important to note that even though Mesh 3 converged using a quadratic polynomial order, the time taken to compute the simulation was extremely long (227 hours) making this simulation not preferred. The same argument is used to avoid simulations with cubic polynomial order solutions for the three meshes since the three simulations needed more than 220 hours for completion. Therefore, the ideal simulation, where both targets, convergence of the solution and the time used to completion are met, can be found using a mesh with an extra 20%, i.e. Mesh 2, of elements using a second order polynomial degree.

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

To save computing time, it is most beneficial to increase the number of nodes only in the areas where more nodes are necessary. If a large section of the part is under a constant stress, then only a few elements will be required. This will save a lot of computing time.

The number of elements must only be increased in areas where the stress changes quickly over a small distance. This could be the area where a load is applied, around a hole, or where geometry is changing. In these areas the stress can change dramatically over a very small distance. It is up to the user to determine where more elements will be required to obtain an accurate solution.

Once the correct refinement has been realized, this model could be used to perform simulations analyzing both, the healthy and the disease state of the MV.

APPENDIX A

Displacement			
	Mesh 1	Mesh 2	Mesh 3
Linear	0.0239338	0.0239596	0.02397
Quadratic	0.0239958	0.024174	0.02421
Cubic	0.0241384	0.0241938	0.02422

Table 3. D_{fe} Data for each simulation.

Error			
	Mesh 1	Mesh 2	Mesh 3
Linear	-3.749026849	-3.794771396	-3.813819591
Quadratic	-3.862702215	-4.287379833	-4.400423524
Cubic	-4.18690057	-4.347967342	-4.43423337

Table 4. Error associated with each D_{fe} measurement.

Dates					
	Mesh 1				
	Start Date	Start Time	End Date	End Time	Hours
Linear	9/25/14	10:01 AM	9/25/14	3:52 PM	5.85
Quadratic	9/22/14	2:23 PM	9/23/14	8:25 AM	18.03333333
Qubic	9/26/14	9:59 AM	10/7/14	4:26 PM	270.45
	Mesh 2				
Linear	9/25/14	10:01 AM	9/25/14	3:50 PM	5.816666667
Quadratic	9/22/14	2:23 PM	9/23/14	6:03 PM	27.66666667
Qubic	9/26/14	10:06 AM	10/6/14	11:41 AM	241.5833333
	Mesh 3				
Linear	10/7/14	10:01 AM	10/7/14	3:55 PM	5.9
Quadratic	10/8/14	2:23 PM	10/9/14	6:21 PM	27.96666667
Qubic	10/10/14	10:01 AM	10/21/14	11:21 AM	265.3333333

Table 5. Computational time for each simulation

Variables	Description
COID	Command ID
ENTYPE	Entity type
ENID	Entity identification number
BCTR	Translational constrains
BCROT	Rotational constraints
CYSIDTR	Coordinate system ID for translational constraints
CSYSIDROT	Coordinate system ID for rotational constraints (negative or 0 for rotation around COG)
TBEG	Activation time
TEND	Termination time
PMETH1	Method used to prescribed the motion (acceleration/velocity or displacement)
DIREC1	Direction of prescribed motion
CID1	ID of a CURVE or FUNCTION defining prescribed motion
SF1	Scale factor
PMETHN	Method used to prescribe motion (acceleration/velocity or displacement)
DIRECN	Direction of prescribed motion
CIDN	ID of a CURVE or FUNCTION defining prescribed motion
SFN	Scale factor for curve ordinate values

Table 6. *BC_motion* **parameter description.**

Command	Generic Description	Specific Description
UNIT_SYSTEM	Command to inform the solver of used unit system.	
PARAMETER	The purpose of this command is to define parameters that can be used inside expressions anywhere in the command file. Brackets are used to mark the beginning and end of an expression.	
INCLUDE	This command is used to merge a file with the input.	Includes the file *mitralmesh.k* which is the mesh to be simulated
INCLUDE		Includes the file *mitral_texture.k* ***
INCLUDE		*2Pistons-Cylinder_ShiftProlonged-NewPlate_AllEnlarged.k*
SET_PART	This command defines a set of parts.	Anterior and Posterior leaflet are assigned the ID = 2000
MERGE_DUPLICATED_NODES	The command is used to merge duplicated nodes	Part 2000 is merged with the plate.
MERGE_DUPLICATED_NODES		The plate and the pipe are merged.
TIME	This command is used to define the duration of the simulated event and to specify parameters controlling the time step size.	Our time step is set to 7.0e-7
OUTPUT	This command contains output parameters, such as output frequency and filter.	
SMS	Selective mass scaling functionality, as described by Olovsson et al. (2005). The critical time step of an element is: $\Delta tc=1+sf\cdot\Delta tc0$, where $\Delta tc0$ is the critical time step without mass scaling.	
GEOMETRY_PIPE	This command is used to define a straight pipe or cylinder in space, with its face center coordinates at (x1, y1, z1) and (x2, y2, z2).	It creates the pipe inside of which the two pistons move and the plate and the mitral valve is positioned
GEOMETRY_BOX	This command is used to define a box in space, with corner coordinates at (x1, y1, z1) and (x2, y2, z2).	Defines the space in which the fluid particles exists and it defines the parts which the fluid particle interact.
PART	The command is used to assign properties to a part or to a range of parts.	
MAT_FLUID	This is a simple fluid model. It can be combined with an equation-of-state for a non-linear pressure-volume relationship.	
SPH_FLUID	Defines SPH fluid geometries/grids and interaction between SPH nodes and structure.	This parameter is defined so that the entire MV interacts with the fluid defined as *GEOMETRY_BOX*
SET_PART	This command defines a set of parts.	Assigns an ID to the entire assembly of MV, pistons, pipe and plate
SET_PART	This command defines a set of parts.	Assigns an ID to only the MV
BC_MOTION	Definition of kinematic boundary conditions.	The pipe is defined as no motion in XYZ directions.
BC_MOTION		The plate is defined as no motion in XYZ directions
BC_MOTION		Piston 1 is defined as no motion in XY direction, and free to move in the Z direction with a velocity defined in *Curve1000Piston1.k*
BC_MOTION		Piston 2 is defined as no motion in XY direction, and free to move in the Z direction with a velocity defined in *Curve1000Piston1.k.*

INCLUDE	This command is used to merge a file with the input.	*Curve1000Piston1.k* which defines the velocity profile of piston 1.
INCLUDE		*Curve1001Piston2.k* which defines the velocity profile of piston 2.
CHANGE_P-ORDER	Change element polynomial order in a selected region of a part or part set.	This parameter is varied from 1..3 or linear to cubic, to increase the p-refinement
INCLUDE	This command is used to merge a file with the input.	*Tissue_material_Models.k* which defines the parameter *mat_tissue_disspersed* (see **Table 3**)
MAT_RIGID	Defines a rigid material.	This parameter is used to defined the pipe to be a rigid body with density 1000 kg/m3
PART	The command is used to assign properties to a part or to a range of parts.	Defines part 1: Anterior leaflet and the material proprieties of this part.
…	…	… *
PART		Defines part 76: Piston 2 and the material proprieties of this part.
CONTACT	Penalty based node-to-surface contact algorithm. Both (slave) nodes and (master) faces are defined from parts or part sets.	

The file '**GMSHtoImpetus.m**' performs the conversion from a GMSH file to Impetus platform.

Table 7. Main.k file description. Summarizes the input parameters defined in main.k. The commands shown in the table are ordered based on the appearance in the *main.k* file. (*) The definition for parts from 2 to 75 are excluded.

The file **'ConnectLines.m'** connect all the points in the MV.

The file **'extractMitral.m'** extracts any part that the user desires.

The file **'main.k'** receives references to **'Stress_Strain_Curves.k'** and **'Metal_Material_Models.k'** also the mesh, and a piston.

The file **'Stress_Strain_Curves.k'** defines the stress and strain curves for the materials.

The file **'Metal_Material_Models.k'** defines the material to be use in the simulation.

REFERENCES

1. d'Arcy, J.L., et al., *Valvular heart disease: the next cardiac epidemic.* Heart, 2011. **97**(2): p. 91-3.

2. Rim, Y., et al., *Evaluation of mitral valve dynamics.* JACC Cardiovasc Imaging, 2013. **6**(2): p. 263-8.
3. Noack, T., et al., *New concepts for mitral valve imaging.* Ann Cardiothorac Surg, 2013. **2**(6): p. 787-95.
4. Su, B., et al., *Numerical simulation of patient-specific left ventricular model with both mitral and aortic valves by FSI approach.* Comput Methods Programs Biomed, 2014. **113**(2): p. 474-82.
5. Lau, K.D., et al., *Fluid-structure interaction study of the edge-to-edge repair technique on the mitral valve.* J Biomech, 2011. **44**(13): p. 2409-17.
6. Modi, P., A. Hassan, and W.R. Chitwood, Jr., *Minimally invasive mitral valve surgery: a systematic review and meta-analysis.* European journal of cardio-thoracic surgery : official journal of the European Association for Cardio-thoracic Surgery, 2008. **34**(5): p. 943-52.
7. Cheng, D.C.H., et al., *Minimally invasive versus conventional open mitral valve surgery: a meta-analysis and systematic review.* Innovations (Philadelphia, Pa), 2011. **6**(2): p. 84-103.
8. Ionasec, R.I., et al., *Patient-specific modeling and quantification of the aortic and mitral valves from 4-D cardiac CT and TEE.* IEEE transactions on medical imaging, 2010. **29**(9): p. 1636-51.
9. Rim, Y., et al., *Mitral valve repair using ePTFE sutures for ruptured mitral chordae tendineae: a computational simulation study.* Ann Biomed Eng, 2014. **42**(1): p. 139-48.
10. Schievano, S., et al., *Percutaneous mitral valve dilatation: single balloon versus double balloon. A finite element study.* J Heart Valve Dis, 2009. **18**(1): p. 28-34.
11. Wenk, J.F., et al., *First finite element model of the left ventricle with mitral valve: insights into ischemic mitral regurgitation.* Ann Thorac Surg, 2010. **89**(5): p. 1546-53.
12. Maisano, F., et al., *An annular prosthesis for the treatment of functional mitral regurgitation: finite element model analysis of a dog bone-shaped ring prosthesis.* Ann Thorac Surg, 2005. **79**(4): p. 1268-75.
13. Stevanella, M., et al., *Mitral leaflet modeling: Importance of in vivo shape and material properties.* J Biomech, 2011. **44**(12): p. 2229-35.
14. Mansi, T., et al., *Towards patient-specific finite-element simulation of MitralClip procedure.* Med Image Comput Comput Assist Interv, 2011. **14**(Pt 1): p. 452-9.
15. Einstein, D.R., et al., *Fluid-Structure Interactions of the Mitral Valve and Left Heart: Comprehensive Strategies, Past, Present and Future.* Int J Numer Methods Eng, 2010. **26**(3-4): p. 348-380.
16. Mansi, T., et al., *An integrated framework for finite-element modeling of mitral valve biomechanics from medical images: application to MitralClip intervention planning.* Med Image Anal, 2012. **16**(7): p. 1330-46.
17. Freed, A.D., D.R. Einstein, and I. Vesely, *Invariant formulation for dispersed transverse isotropy in aortic heart valves: an efficient means for modeling fiber splay.* Biomech Model Mechanobiol, 2005. **4**(2-3): p. 100-17.
18. Freed, A.D. and T.C. Doehring, *Elastic model for crimped collagen fibrils.* J Biomech Eng, 2005. **127**(4): p. 587-93.